

Demo Project for Index Function

Table of Contents

1. Overview and Operation
2. Setting Up the Screen
3. Object

1. Overview and Operation

The EB8000 provides 32 index registers for users to use addresses more flexible. Via index register, users can update object's read / write address without changing content of the object under the machine in operation.

Address Tag Library

Customized System

No.	Address tag name	PLC name	Address type	Device type	Address	Read/Write
228	L.W-9200 (16bit) : address index 0	Local HMI	Word	L.W	9200	Read/Write
229	L.W-9201 (16bit) : address index 1	Local HMI	Word	L.W	9201	Read/Write
230	L.W-9202 (16bit) : address index 2	Local HMI	Word	L.W	9202	Read/Write
231	L.W-9203 (16bit) : address index 3	Local HMI	Word	L.W	9203	Read/Write
232	L.W-9204 (16bit) : address index 4	Local HMI	Word	L.W	9204	Read/Write
233	L.W-9205 (16bit) : address index 5	Local HMI	Word	L.W	9205	Read/Write
234	L.W-9206 (16bit) : address index 6	Local HMI	Word	L.W	9206	Read/Write
235	L.W-9207 (16bit) : address index 7	Local HMI	Word	L.W	9207	Read/Write
236	L.W-9208 (16bit) : address index 8	Local HMI	Word	L.W	9208	Read/Write
237	L.W-9209 (16bit) : address index 9	Local HMI	Word	L.W	9209	Read/Write
238	L.W-9210 (16bit) : address index 10	Local HMI	Word	L.W	9210	Read/Write
239	L.W-9211 (16bit) : address index 11	Local HMI	Word	L.W	9211	Read/Write
240	L.W-9212 (16bit) : address index 12	Local HMI	Word	L.W	9212	Read/Write
241	L.W-9213 (16bit) : address index 13	Local HMI	Word	L.W	9213	Read/Write

* Users can import MTS00 tag to represent the address.

2. Setting Up the Screen

1. Create ASCII and Numeric Input objects from LW0~19. These objects must use the same index register (INDEX 0).

The screenshot shows the 'ASCII Input Object's Properties' dialog box with the 'General' tab selected. The 'Read address' section is highlighted with a red box and contains the following settings:

- 1. PLC name: Local HMI
- Device type: LW
- Address: 0
- System tag:
- 2. Address format: ddddd [range: 0 ~ 10255]
- Index: INDEX 0 (16-bit)
- Index register:
- No. of words: 5

The 'Notification' section at the bottom has the 'Enable' checkbox unchecked.

The screenshot shows the 'Numeric Input Object's Properties' dialog box with the 'General' tab selected. The 'Read address' section is highlighted with a red box and contains the following settings:

- 1. PLC name: Local HMI
- Device type: LW
- Address: 5
- System tag:
- 2. Address format: ddddd [range: 0 ~ 10255]
- Index: INDEX 0 (16-bit)
- Index register:

The 'Notification' section at the bottom has the 'Enable' checkbox unchecked.

2. Duplicate the first step again. To create ASCII and Numeric Input objects from RW0~19. These objects must use the same index register (INDEX 1).

ASCII Input Object's Properties

General | Data Entry | Security | Shape | Font | Profile

Description : _____

Use UNICODE Reverse high/low byte

Read address

1. PLC name : Local HMI

Device type : RW

Address : 0 System tag

2. Address format : ddddd [range : 0 ~ 65535]

Index : INDEX 1 (16-bit) Index register

No. of words : 5

Notification

Enable

Numeric Input Object's Properties

General | Data Entry | Numeric Format | Security | Shape | Font | Profile

Description : |

Read address

1. PLC name : Local HMI

Device type : RW

Address : 5 System tag

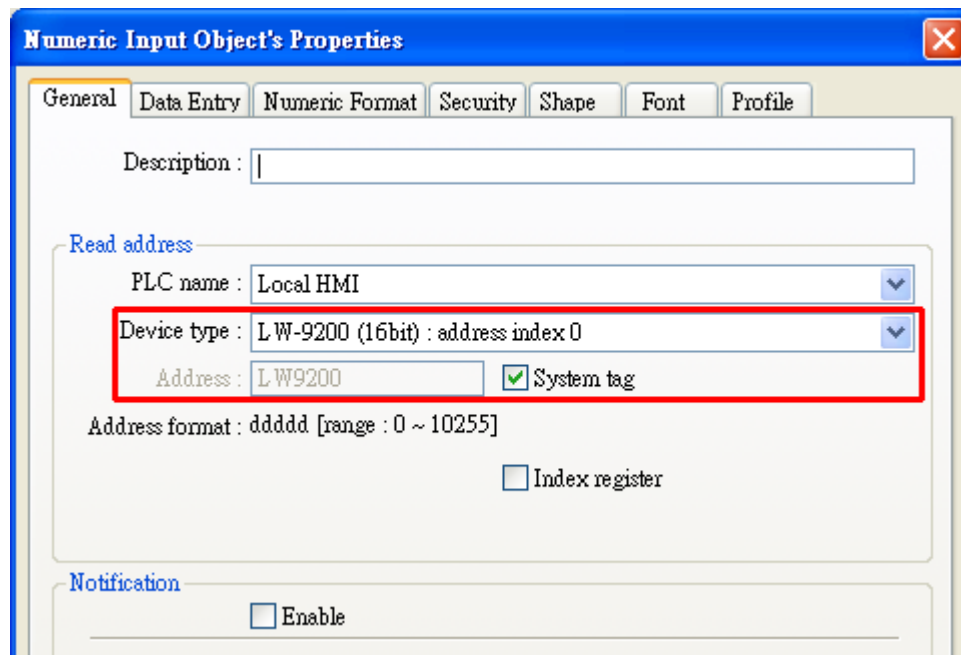
Address format : ddddd [range : 0 ~ 65535]

2. Index : INDEX 1 (16-bit) Index register

Notification

Enable

3. Create Numeric Input Object and set up the address as system tag (LW9200). User can use this object to change the LW0~19 address offset.



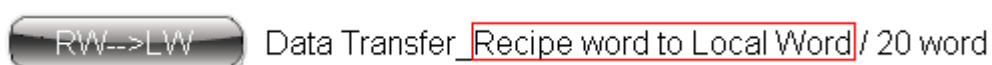
4. User can create two Set word object to switch the address offset of Index 0. (Set Style: JOG-/JOG+ ; address offset: 20 words)

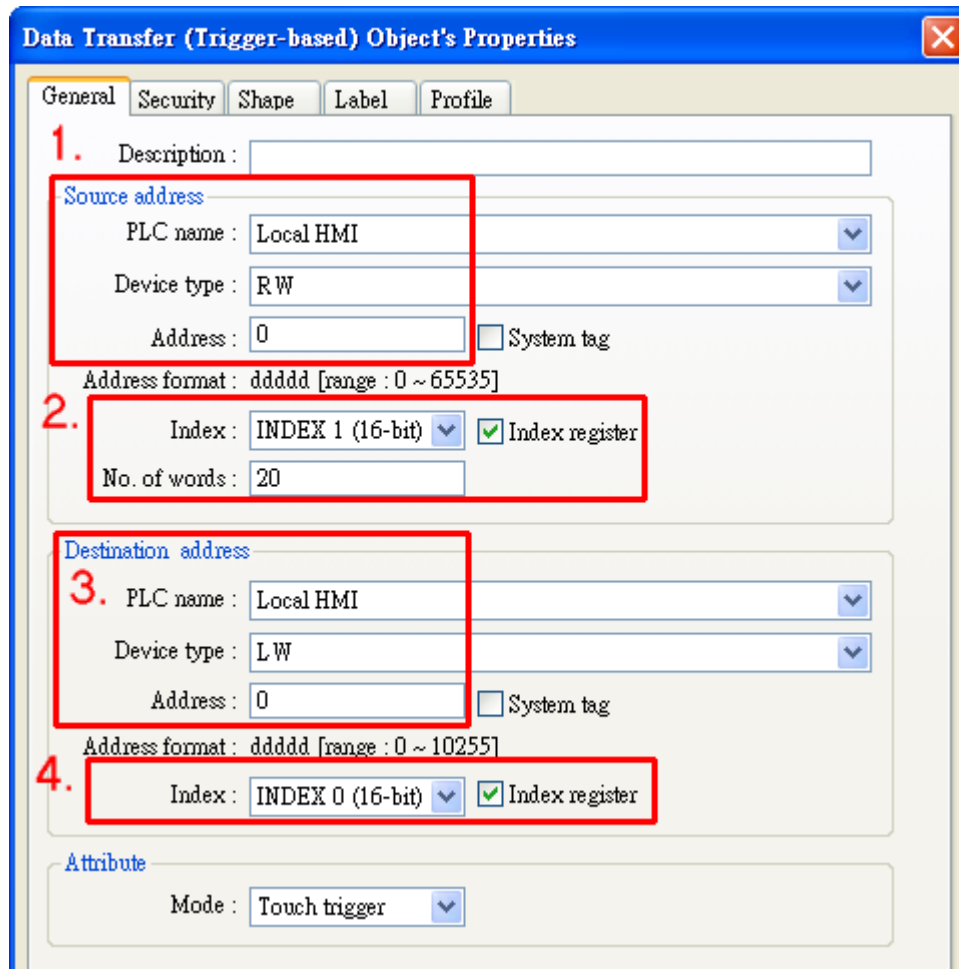


5. Duplicate the above step again. Create two Set word object to switch the address offset of Index 1. (Set Style: JOG-/JOG+ ; address offset: 20 words)

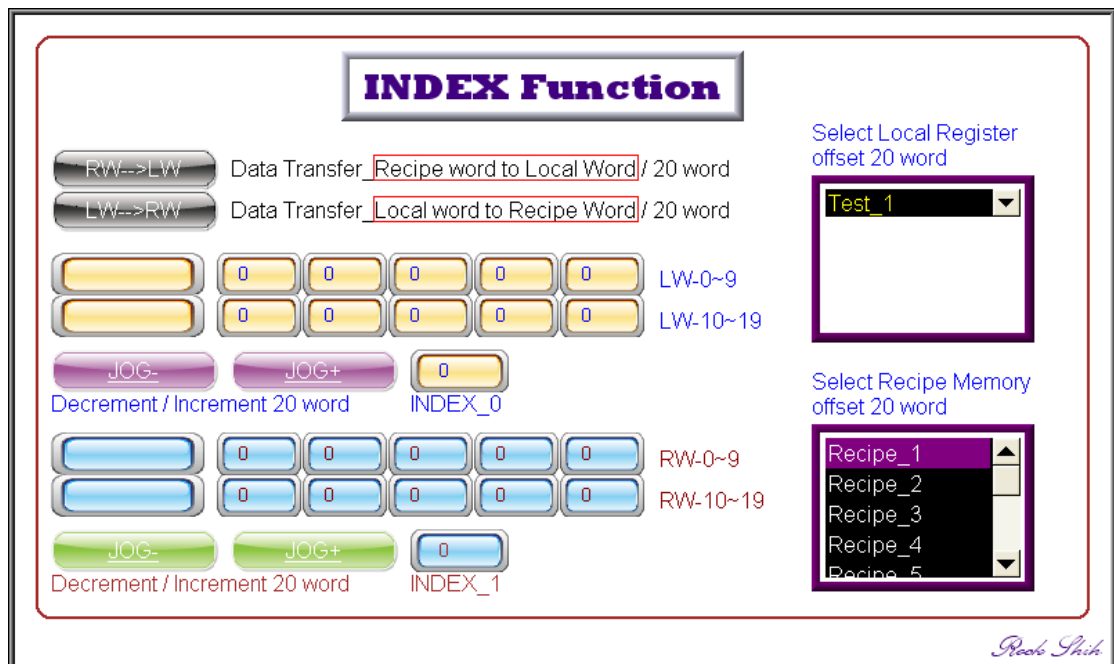


6. Create two [Data transfer] objects to transfer the data from source (RW) to destination (LW). For example: RW→LW, the RW register must use the index 1; on the contrary the LW need use the index 0.





7. User can transfer the data from RW (LW) to LW (RW) by [Data transfer object]. And also use the [set word object] or [Option list object] to choose the offset address of these index registers.



3. Object

The objects are used in this demo project as the following area.

Object	ID	Detail
Data Transfer	RP0/RP1	Transfer the RW/LW register data.
ASCII Input	AE0	RW0~4
	AE1	RW10~14
	AE2	LW0~4
	AE3	LW10~14
Numeric Input	NE0~9	LW5~9 /LW15~19
	NE10	INDEX 0
	NE11~20	RW5~9 /RW15~19
	NE21	INDEX 1
Set word	SW0	LW register (JOG-)
	SW1	LW register (JOG+)
	SW2	RW register (JOG-)
	SW3	RW register (JOG+)
Option List	OL0	Select Local Register
	OL1	Select Recipe Register